AndPskmail User's Manual Version Alpha-0.93

John Douyere (VK2ETA) on 21-Oct-2011

1 Acknowledgements

I want to thank Rein Couperus (PA0R) for his relentless development of new Pskmail features and solutions. Without his drive and persistence Pskmail would not simply be, or not be what it is today, a feature reach bridge with the Internet with very practical use for mobile and portable stations needing access to key services.

I want to thank Rein and Per Crusefalk (SM0RWO) for their support and the jPskmail Java code under open source license. I believe this demonstrates the benefits of such approach for our hobby. By the way, Rein still accepts a beer as a donation. Thank you to Per for hosting the subversion system which allows the development team to efficiently share the source code of the applications.

I also want to thank David (Dave) Freese (W1HKJ) and team for Fldigi which code under GPL was the basis of the modem's implementation in AndPskmail and specifically Dave who supported me when developing the PSK-Robust modes for Fldigi.

Thank you to Franz-Josef Maas (DB3CF) who translated the PSK modem code from Fldigi into Java, providing a strong starting point for the PSK modes implementation in this development.

Finally, thank you to the Pskmail community for their ongoing feedback and support.

2 Introduction

The main objective of this development is to provide an ultra-portable, as in backpack or bicycle for example, Pskmail client solution.

Rigs like the Yaesu's FT-817, Icom's 703 and the soon to be released Elecraft's KX3 would be the typical companions of the Android phone or tablet running AndPskmail, thereby providing a low weight, low energy requirement, access to most Pskmail services.

The situations targeted are where the operator, most likely being in an area outside of mobile phone coverage, is primarily concerned in reporting his situation and GPS position, plus raising alert if necessary. Secondary considerations will be exchanging emails and accessing web pages information like weather forecasts.

Other uses would be in energy constrained situations like Maritime Mobile where the device could report position on a regular basis (in typical APRS fashion) without the battery drain and/or space requirement of a laptop or netbook.

Finally, since speaker to microphone audio coupling is showing good results, for "quick" transactions when mobile or maritime mobile and wanting to send a simple status back to the server without setting up the laptop. Just run an APRS beacon or

message in front of the transceiver's microphone. This could be applicable for VHF/UHF FM operations too.

3 Why Android?

The key considerations where the re-use of the code available under open source GPL license in jPskmail, the PC version that Rein and Per developed, AND the wide availability of devices. About 60 to 70% of the Java code used in AndPskmail is directly from jPskmail, dramatically reducing the development and testing time of the project.

Also, the frequent presence of an integrated GPS receiver and hands free connection (meaning speakers and microphone connections), the plethora of built-in services in the Android OS and it's open source development kit approach makes it a very attractive platform to develop a Pskmail client for ultra-portable situations.

It is designed to run on on various devices (phones with limited screen space as well as tablets) from Android version 2.1 onwards, although I am yet to test it on a tablet's larger screen. It works both in portrait and landscape mode.

4 Typical hardware configuration

Android phone or tablet. Integrated GPS would be a plus for automatic position reporting (Beacons).

Simple audio coupling (speaker to microphone) is giving good results provided the environment is not too noisy.

New: **I have also successfully tested a Bluetooth interface to TX and RX the audio to/from the radio without wires. This represents a very handy and lightweight set-up. A simple low cost Bluetooth handsfree car kit (the type that clips on the sun visor) is used as the interface but some simple soldering is required. Schematics and pictures coming soon.**

I have also made an interface cable without galvanic isolation between the FT-817 and an HTC Desire Android phone. Galvanic isolation is not critical since there is no ground loop as the power is inside the device. I found that simply inserting ferrite beads on the audio cable between the phone works relatively well.

The only issue with a non-isolated interface is that the touch screen of these devices may not respond when RF is present. This means timing the screen input during RX periods.

For the rig I use an FT-817 and connect to the microphone socket as the VOX only works in voice mode (USB/LSB) not in digital mode (DIG).

Antennas are of course what you use normally. I use a Fuchs antenna and tuner. This is a 41 metres wire end-fed antenna that tunes to all bands with good efficiency, but I have used G5RVs and car mounted antennas effectively too.

So in summary and as an example, here is my setup: Android phone (HTC desire, but I have also test on a Huawei Sonic), simple cable from the phone to the rig's microphone and speakers, FT-817, Lithium battery pack (low weight), Fuchs antenna (41M long) and tuner. Since I also use the phone for digital mapping (OruxMaps) when "out bush" there is no additional weight required.

Compatibility: has been tested on HTC EVO 4G, HTC Desire, HUAWEI Sonic. It should run on any device with Android 2.1 or later.

5 What does it do?

Basically most of what the fully featured version on the PC does. In no particular order:

- In unconnected mode: send position reports (using the integrated GPS if there is one), Pings, Inquiries, Link to server and APRS messages.

  I use this last feature in conjunction with an Email-To-SMS provider, allowing me to send SMSs to any mobile phone without having to go through the process of a connected session. Very handy in my experience.

- In connected mode: List email headers, send and receive emails in compressed or uncompressed format, request all the services from the server like fleetcodes, WWV propagation data, GRIB files and last but not least access to any web page text, again in compressed or uncompressed format.

  If using the compressed format, AndPskmail will also handle the uncompleted transactions and will resume the uploads or downloads from where they were left at during the previous sessions. The choice is available to the operator to reject or discard uncompleted transaction.

  The new STOP Transaction allows to receive just the minimum amount of data necessary by manually stopping the transaction when the critical information is received on the portable device. This applies to emails and web pages in uncompressed format, as the operator can see the text being downloaded in the terminal screen

  New: Connections are now possible in any PSK mode from PSK31 to PSK500 ***and PSK125R to PSK500R*** and any mode in-between, with automatic transmission and reception of the mode ID by RSID (Reed Solomon ID).

  This allows for the modes to be changed automatically by the server and this independently in RX and TX to adapt to the conditions and noise.

  This requires server 1.4.12 or later. Earlier servers will work with this

client's version but will only allow modes PSK125 to PSK500 and PSK250R to PSK500R.

What is NOT available at this point (but may well be in the development pipeline, see below):

- Modes other than PSK31 to PSK500 and PSK125R to PSK500R.
- Files Uploads and Downloads (possible but requires typing commands by hand).
- TTY mode for ARQ message exchanges.
- Access to Local mail (non-internet email) on the server.
- Telnet access through the server (possible but requires typing commands by hand)..

6 Installation

Download the AndPskmail.apk file from the website at:

http://www.pskmail.org/AndPskmail.html

And save it to the SD card, either from within the phone or on a PC to which the phone is connected to and has the USB driver installed for that phone to allow access to the SD card content from the PC.

From there use either a file explorer on the device or the automatic install function (like in the Huawei Sonic) to install the .apk file.

You may have to allow non-Android market application to be installed for this to work. This is found under Setting / Applications / Unknown Sources.

You will be asked to allow the application to access your position and the SD card. This is necessary for AnPskmail for position reporting and all the working files are stored directly on the SD card under the “pskmail” directory.

In your applications list you should now have a Pskmail logo called AndPskmail.

7 Usage

I have tried to maintain a similar philosophy of operation with the PC version.

When you launch the AndPskmail application you are presented with the terminal screen with the typical buttons to Connect/Disconnect, list pending transactions, send mail stored in the Outbox, Abort the session or Stop the current transaction (but remain connected).

The other screens available are:

- Email Headers screen, where the list of headers is displayed and all the email actions are performed from . From this screen we also access the New Mail entry screen plus the listing of Inbox and Outbox mails and their display.

- Modem screen, where all the received and transmitted characters are shown and where the modem can be turned ON and OFF to conserve batteries. The squelch is also adjusted there and the waterfall is accessible from that screen.

- APRS screen, where only APRS data received from the servers is shown and where Beacons, Pings, Inquiries, Link-to and APRS messages can be sent.

- Info screen, where all the requests for information including web pages are made from.

- About screen. The usual, accessible via the menu only.

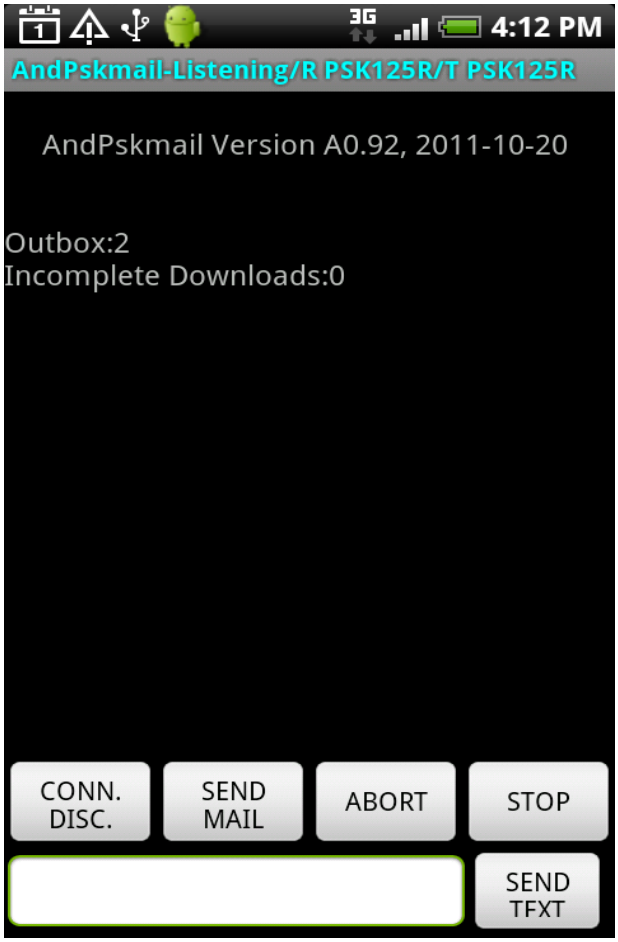
Movement between these screens is done via the menu (see below).

From the menu we also access the exit function, the user's preferences and the list of clean-up options (clear mail headers, Inbox, Outbox etc...).

One preference (option) that is new is the option to connect with list of mode or not. This should be selected for servers version 1.4.0 and up. For others, de-select that option.

8 Screens description

8.1 Terminal Screen



At the top of the screen the status line shows the session's status: Listening (meaning not in a connected session), “Connecting”, “Connected” and “Disconnecting”.

New: **In the title bar we also now display the Rx and Tx modes and the RX / TX status (Light Blue = RX, Yellow = TX, White = Modem OFF).**

The received text can be scrolled up and down.

The “CONN. DISC.” button send a Connect command when the status is “Listening” and a Disconnect command when the status is “Connected”.

The server's call sign is set in the preferences.

New: **“Pending transactions”, for listing incomplete downloads and messages in the Outbox, has been moved to the menu list as it is not used that often. Therefore the button has disappeared from the previous version.**

“Send Mail”, is a duplicate of the same button in the email screen and is here for

convenience. It send outstanding emails from the Outbox, resuming previous downloads if possible to minimise traffic.

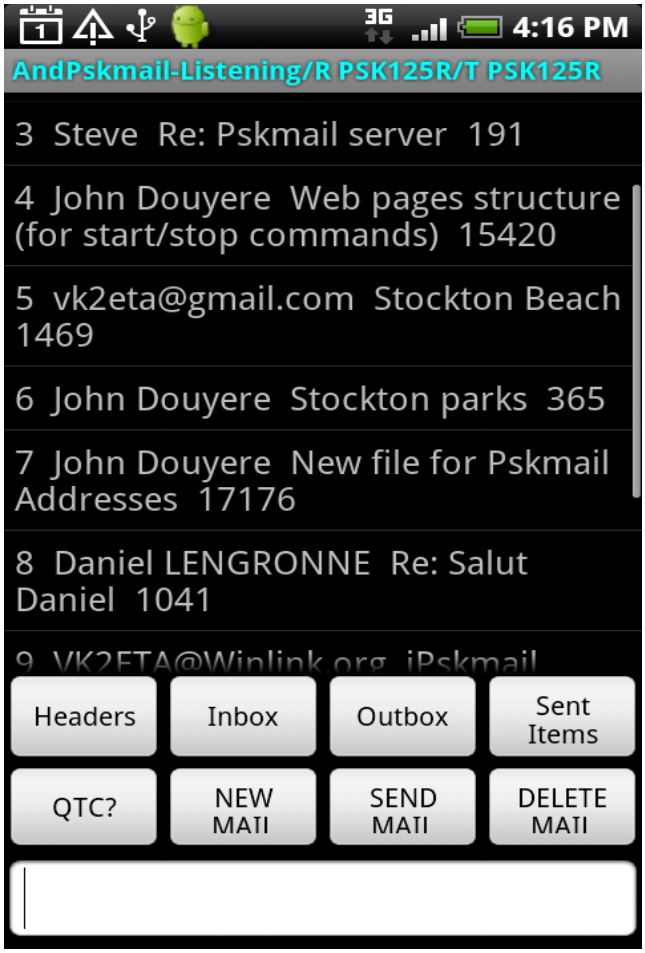
“Abort” to cancel abruptly the current session. It is preferable to use the Disconnect button above, but can be useful in case the link is lost and the server still tries to contact the client.

“Stop” is a new command that stops the current transactions (email or web page download for example) but does not disconnect the session. This is useful for preventing an unwanted transmission to be processed before we can move on to the next request.

This can be used in uncompressed mode as well for stopping the remaining unwanted portion of an email or web page to minimise TX and RX time when operating from batteries.

Also note that the AndPskmail application will continue to run even if the phone's screen is switched off so that batteries can be conserved during the data exchange.

8.2 Email Screen



The data entry box at the bottom of the screen is used to type in email numbers.

The list is can be scrolled up and down.

New: **Headers, Inbox, Outbox and Sent Items buttons cycle the screens between these folders. The sent items button is not implemented yet.**

Note: the Inbox display can be slow if the Inbox file is more than a few kilobytes long. This will be worked on next.

New: **A "long press" (i.e. press and hold) on an email header will request that email to be downloaded. The READ button has been removed altogether.**

New: **A "long press" on any of the other screens' list (Inbox and Outbox only at present) will display the email in question in a pop-up window. Press the return button to return to the list.**

QTC is to request email headers. ***New:*** **Without data entered at the bottom, it will request new headers past the last on in the list. If a number is entered it will request headers from that number.**

The download will be compressed or uncompressed depending on your selection in the preferences

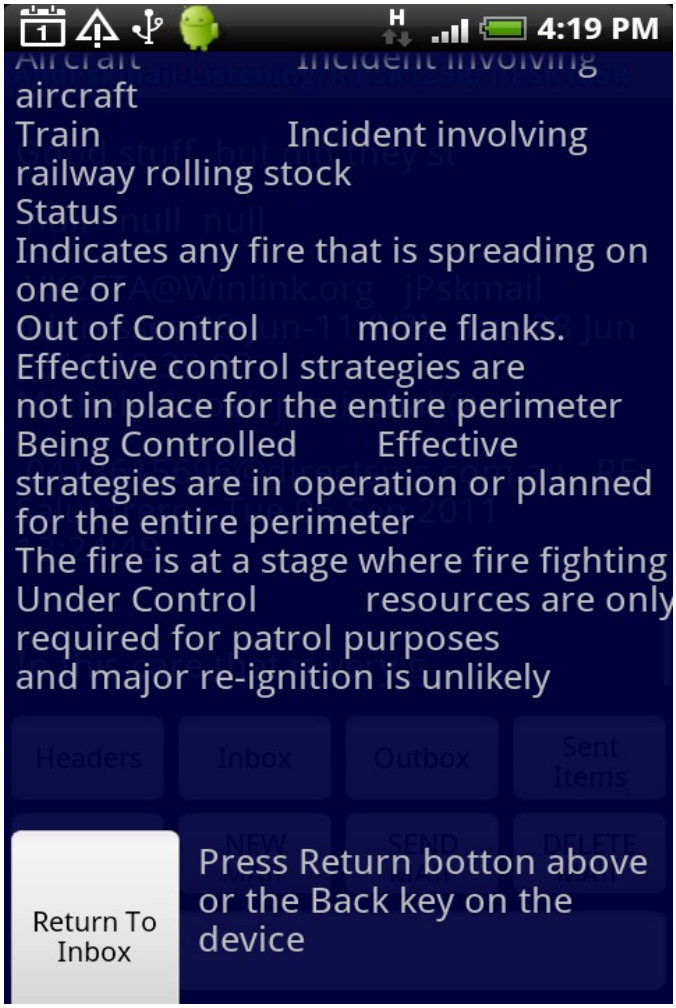
New Mail will open a new dialogue for typing-in the new email.

Send Mail, as per the terminal button of the same name.

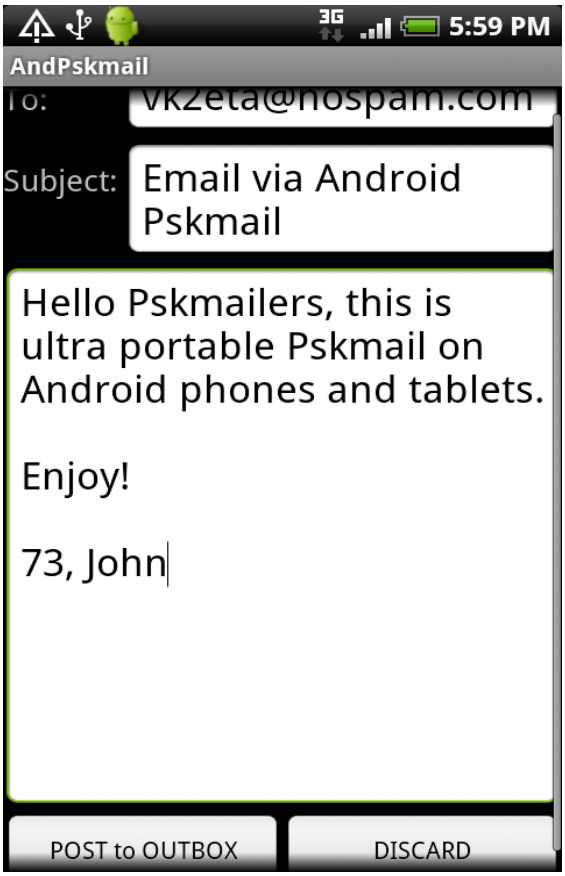
Delete, request deletion of your email as per the supplied number below.

Be mindful that email deletion results in all emails of higher position to be shifted down by one, so the numbering in the list above becomes obsolete with the first delete action.

An example of an email pop-up display:



8.3 The New Mail screen



This screen is “scrollable” in it's entirety, meaning that you can press and move the screen up and down, would the text entered cause it to become bigger than the device's screen.

Useful when trying to find the Post to Outbox button when the text expands.

New: **A problem has been fixed: on opening of this window the keyboard didn't show up even if clicking on the To: field.**

Note: navigating away from that screen, like pressing the device's home button, will result in the loss of the text typed in. This will be resolved.

8.4 The Modem Screen

To monitor the data received and sent by the software modem and see an indicator of CPU load.

The top (normally green) bar represents the CPU load with zero load on the left and 100% on the right. This is an indication of how much reserve the CPU has, rather that a true load indicator.

The bottom bar (2 overlaying bars in fact) represents both the squelch level and the current received signal quality. The brighter one is the signal quality and the darker one the squelch. The signal quality moves in a comparable way to the Fldigi green bar on the right hand side of the waterfall.

New: **The squelch up and down increases/decreases the squelch value and stores it for later use so that it stays until changes even across sessions. The dark green bar represent the squelch level.**

New: **The mode up and mode down buttons will cycle either through your custom list of modes as set in the preferences OR through the whole list of modes available is the "Use custom list of modes" preferences is not set.**

Modem OFF stops reception and further processing of incoming audio. This is to conserve processing and therefore batteries.

New: **This button can now be pressed repetitively as it should not crash the application (please test).**

New: **The Mode change is now automatic on return from the preference screen.**

New: **The Tune button produces a 3 seconds tune at the current set level of the modem's volume. Handy for testing SWR and tuning.**

New: **The Stop TX button is for immediate silencing of the modem in case of unwanted transmission.**

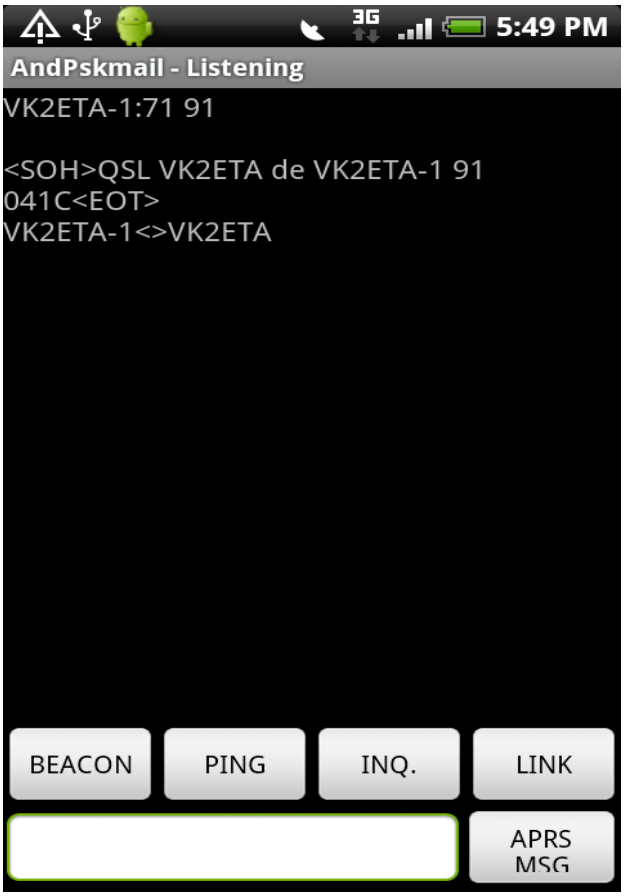
The decoded RSID mode and centre frequency are displayed in the Modem screen.

New: **The waterfall can now be displayed by pressing the W.FALL on/off button. See screen copy below.**

Be aware **that the CPU load increases when displaying the waterfall and on some devices doing so while on PSK500R can lead to losses of data AND making RSID not detectable.**

Use this waterfall display sparingly on these devices.

8.5 The APRS screen



Upon entry to this APRS screen, the GPS will be automatically switched on, if not disabled at the device level by the operator. See phone/tablet “Settings” / “Location” options.

The text displayed in this screen is only APRS type of received traffic.

The text entry at the bottom of the screen is for APRS messages (short unconnected emails or pure APRS messages).

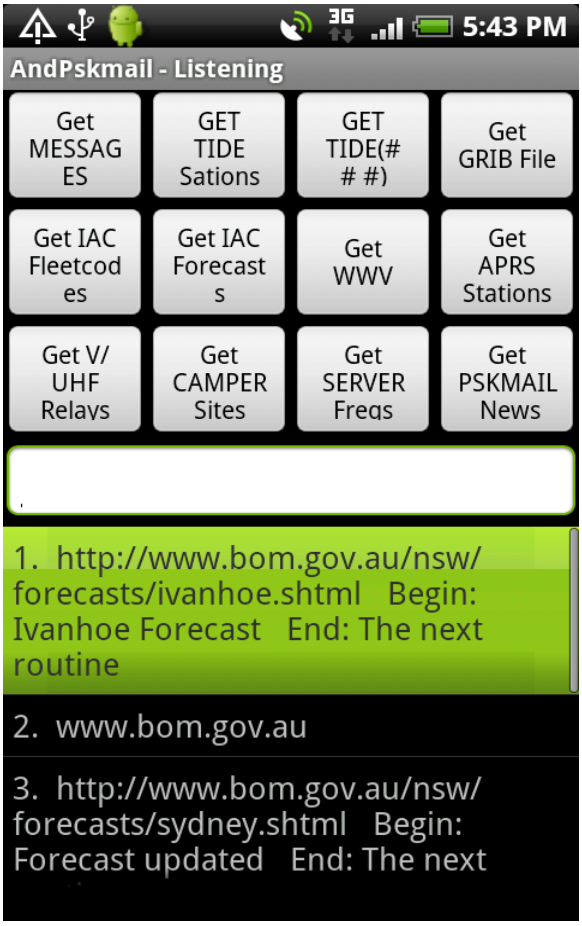
The server's call sign is the one from the preferences.

The Beacon button will request the position from the built-in GPS and send the “Last known position”, which could be old data if the GPS has been just switched on.

Better handling of old position data will be integrated in future versions. Ideas are welcomed in that regard.

Note that battery consumption is much higher when the GPS is ON, therefore it is better to move to another screen when we do not need to send APRS data to conserve batteries. A better GPS on/off scheme is in the pipeline for that effect.

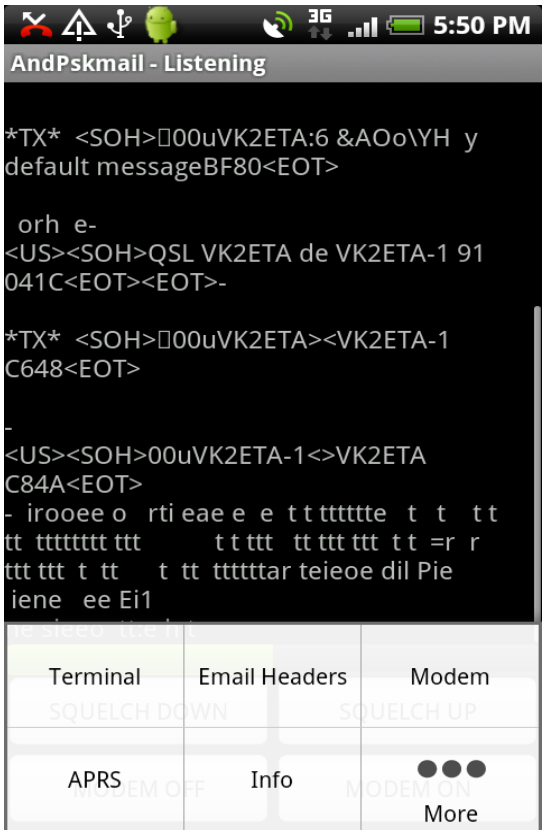
8.6 The Info Screen

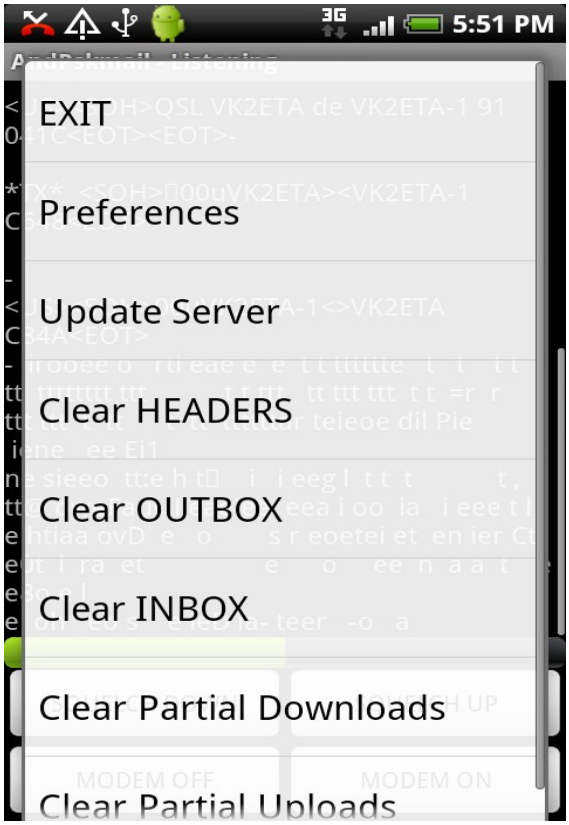


The text entry in the middle of the screen is for specifying Tide station number, gps positions for GRIB file requests, V/UHF relays nearby and European camper sites.

To select a web page for download press and hold an entry in the scrollable list of web pages (“long press” in Android talk).

8.7 The Navigating Menu





Upon pressing the menu button on the device, the menu is displayed as per the left screen copy above. Pressing the “More” button will bring the remaining part of the menu as a list as seen on the right screen copy.

The first section is for navigating between screens.

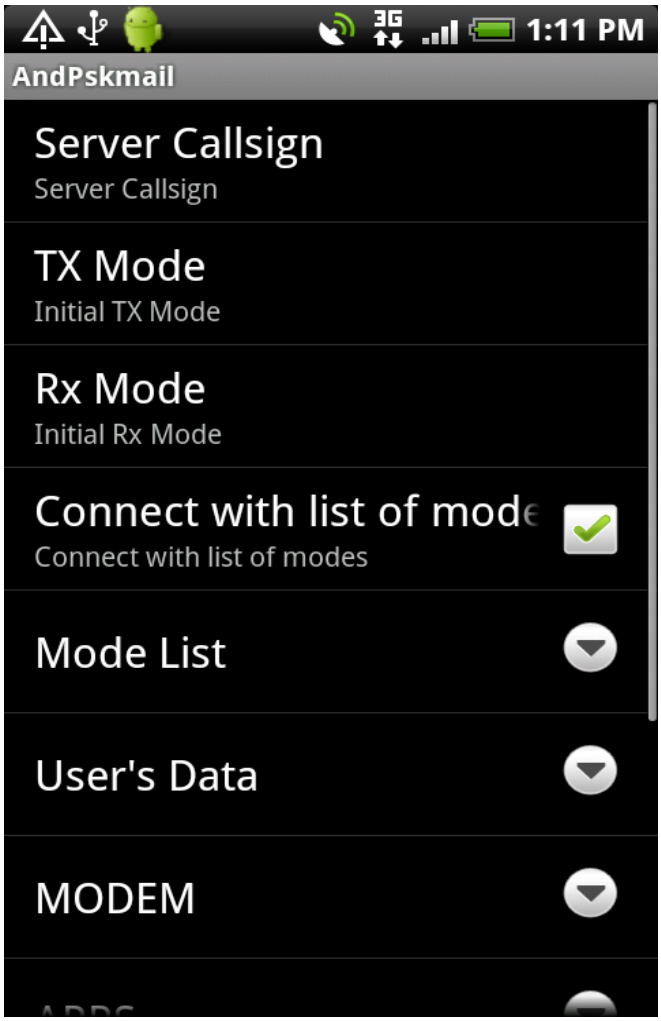
In the second section, the exit button returns to the previous displayed application or main device screen, and stop the software modem.

The other buttons are self-explanatory and similar to the jPskmail client.

Update server will determine the server's version and adjust it accordingly (with or without session password).

New: **There is now an About screen which displays the version and some information as usual.**

8.8 The Preference screen



Preferences are now grouped in preference screens (accessed via the down arrow) with the most common preferences on top of the list.

This is a scrollable list of options.

Pretty much like the list on the jPskmail PC client with a few exceptions:

- Modem volume and audio frequency since the modem is integrated in the application. Remember, for the time being, these parameters require a modem restart.

- Delay for switching modem off: not used yet (upcoming auto off feature).

- “Connect with list of mode” for connecting to servers 1.4.0 and beyond. Fill-in the mode list (next one down in the preferences). See next screen copy for explanation about the list of modes.

- Beacon status. This is the default beacon status text used if no text is entered in the APRS text entry field.

*The "M*ode List" allows the construction of a custom list of modes that the server will choose for the link. Tick the modes wanted.

The modes are ordered in ascending order of average robustness / descending order of speed.

Note that PSK500R can be very CPU demanding. If your device is consistently close to 100% CPU when in this mode (the top bar in the Modem screen), then it is better not to use it as it will result in missed characters requiring numerous re-send of data blocks. Select PSK250 and/or PSK250R instead (as below).

Afterwards, ensure that the TX and TX modes in the preferences are included in this list before connecting. Otherwise the program will choose the slowest mode in the list as the start-up mode for TX and RX.

9 Problems resolved in A0.93

9.1 The mode change now occurs automatically when returning from the preference screen. There no need to stop and restart the modem. Even better use the mode up and down buttons in the modem screen.

9.2 Pressing the Abort button when the application was trying to send a status packet was causing distorted audio or crashing the application.

Now, if currently TXing the program will ignore the input and ask to press the button again during RX.

9.3 The screens are now limited to a maximum of 60,000 characters. Beyond this the text is lost! This resolves the issue of the displayed screen keep growing indefinitely until they reach their maximum size resulting in a locked display.

9.4 The keyboard now show automatically upon entry to the new mail screen. It was necessary to select the subject or main email body section to bring-up the keyboard.

9.5 Resolved the status of the connection (listening, connecting...etc) changing randomly at times and not reflecting the real status. This was related related to a change in screen orientation from portrait to landscape or vice-versa.

10 Know bugs and quirks in version Alpha 0.93:

10.1 Remember this is ALPHA software and is released with known bug for “first impressions and comments” mainly.

10.2 The list of headers is not refreshed automatically after a QTC command is processed. Re-calling the Email Header's screen will refresh it. This will be fixed.

10.3 When creating a new mail, navigating away from the screen before performing a “Post to Outbox” will result in loss of data entered. Will be fixed soon.

10.4 Not a bug per say but and annoyance: the display of the inbox list of mails is slow and can take 20 seconds for a 50KByte inbox file. This will be resolved.

11 What is in the development pipeline?

11.1 Bluetooth audio interface. To be completely wire-free between the Android device and the radio. ***Device modified, bench testing done, field testing in progress.***

11.2 Access the device's address book when composing emails.

11.3 Access the device's phone list when composing APRS messages (for selecting phone numbers for sending SMSes). Details of email-to-SMS provider to be stored in preferences for completing the email address after the phone number.

11.4 TTY ARQ text and file exchange (later).

11.5 Implement a Sent Item folder and list.

11.6 “Swipe” gesture management for “flicking” between screens. This will be easier and more natural than using the menu for changing screens.

11.7 “Double tap” in terminal or APRS to bring the Modem monitor information to see what is happening at the modem level (useful in case of bad reception).

11.8 User confirmations for critical buttons like DISCONNECT, ABORT, STOP, DISCARD Email to avoid errors.

11.9 Auto beacon and auto-link at given intervals.

11.10 Automatic switching of RX modem on and off to conserve batteries: when connecting, will go on and off when disconnected. Will go on for a selected period (e.g. 2 minutes) when sending an APRS message to allow for the server's reply. Useful with the auto-beacon, auto-link feature.

11.11 Summoning the server to a new frequency. Rig control will be manual of course.

11.12 Extended lead-in (option) when Txing for audio coupling to allow enough time for the operator to key-in the transceiver's PTT.

11.13 New modes, in order of priority: THOR (8,11 and 22) or MFSK 16 and 32. These are time consuming development tasks as the CPU and language constraints mean that the code has to be redeveloped and optimised for these devices.

11.14 Handling of no GPS available (or disabled by operator) with option to prompt for position or use the pre-recorded position.

11.15 Copy/Paste sections of the terminal's data for storage in a text file on the SD card (useful when downloading web pages and the information need to be kept for later).

11.16 Sending of attachments with emails.

11.17 Review the overall application flow to for ease of use and consistency with the other Android applications like using the back button versus “flicking between screens”. This will be based on user's feedback.

11.18 Handling of Files uploads and downloads via a screen. At present this is possible but the commands have to be keyed in the terminal window.

11.19 Handling of local mail on server. Again this is available now but requires typing of commands in the terminal window.

11.20 That's all for now :)